Tuan-Anh *et al. BMC Bioinformatics* (2017) 18:100
DOI 10.1186/s12859-017-1517-z

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

Novel methods to optimize gene and statistic test for evaluation – an application for *Escherichia coli*

Tran Tuan-Anh[1], Le Thi Ly[2], Ngo Quoc Viet[3] and Pham The Bao[1*]

Abstract

Background: Since the recombinant protein was discovered, it has become more popular in many aspects of life science. The value of global pharmaceutical market was $87 billion in 2008 and the sales for industrial enzyme exceeded $4 billion in 2012. This is strong evidence showing the great potential of recombinant protein. However, native genes introduced into a host can cause incompatibility of codon usage bias, GC content, repeat region, Shine-Dalgarno sequence with host's expression system, so the yields can fall down significantly. Hence, we propose novel methods for gene optimization based on neural network, Bayesian theory, and Euclidian distance.

Result: The correlation coefficients of our neural network are 0.86, 0.73, and 0.90 in training, validation, and testing process. In addition, genes optimized by our methods seem to associate with highly expressed genes and give reasonable codon adaptation index values. Furthermore, genes optimized by the proposed methods are highly matched with the previous experimental data.

Conclusion: The proposed methods have high potential for gene optimization and further researches in gene expression. We built a demonstrative program using Matlab R2014a under Mac OS X. The program was published in both standalone executable program and Matlab function files. The developed program can be accessed from http://www.math.hcmus.edu.vn/~ptbao/paper_soft/GeneOptProg/.

Keywords: Gene optimization, Neural network, Bayes' theorem, Euclidean distance, Codon usage bias, Highly expressed gene

Background

Since Paul Berg and Peter Lobban each independently proposed an approach to generate recombinant DNA in 1969 – 1970, recombinant protein has become a widespread tool for both cellular and molecular biology. For instance, in 2004, more than 75 recombinant proteins were used as medicine and more than 360 pharmaceuticals based on recombinant protein were under development [1]. Moreover, Elena's study indicated that the global market of industrial enzymes exceeded $4 billion in 2012 [2]. In the future, this figure can be raised considerably thanks to the applications of synthetic biology tools which will improve the productivity of recombinant proteins production.

The increment in recombinant protein productivity reduces a significant production cost, so it might dramatically raise profits. In order to improve the productivity, several aspects can be optimized such as purification process, culture medium and genetic materials (including operator, promoter, and gene). In this study, we only focus on gene optimization.

Introducing native genes into a host can cause incompatibility of codon usage bias, GC content, repeat region, Shine-Dalgarno sequence with host's expression system. The yields can fall down significantly [3–7]. In a culture medium, synonymous genes, which share the same operator and promoter, can be expressed at different levels. The synthetic codon optimized gene results in protein level that were ~2 to 22 fold greater than the amounts reported for the native genes [8–12]. A gene optimization program based on machine learning approach and experimental data can handle redesign task rapidly instead of using "brute

* Correspondence: ptbao@hcmus.edu.vn
[1]Faculty of Mathematics and Computer Science, VNUHCM-University of Science, 227 Nguyen Van Cu Street, District 5, Ho Chi Minh City, Vietnam
Full list of author information is available at the end of the article

force" method, which consume more significant times than other resources.

The foundation of gene optimization is a phenomenon of codon synonym. A codon is constructed by three ribonucleotides, so there are $4^3 = 64$ codons (there are 4 kinds of ribonucleotides: A – Adenine, U – Uracil, G – Guanine and C - Cytosine). However, 61 types of codons can code for only 20 kinds of amino acids. This means there must be several amino acids encoded by at least two codons. If one amino acid is coded by several codons, these codons are called synonymous codons. Moreover, codon usage is diverse from organism to organism [3, 13–15]. Generally, genes having compatible codon usage bias with host's expression system are usually highly expressed in translational levels.

The aim of gene optimization program is to indicate which synonymous genes can give higher yield by using variety of approach including one amino acid – one codon (JCAT, Codon Optimizer, INCA, UPGene, Gene Designer), randomization (DNA Works), hybrid (DNA Works), Monte Carlo (Gene Designer), genetic algorithm (GASSCO, EuGene), etc. [16–22]. In some case, one amino acid – one codon method which replaces rare codons by the most preferable usage codons can result in worse protein expression as reported in many past studies [11, 23–26]. Yields of genes redesigned by randomization method are greater than yields of native genes, yet the result is uncertain and we cannot predict expression level until experiment finished [11, 20]. Genetic algorithm and Monte Carlo method with linear target function seem more reasonable than other reported methods. However, parameter estimation has been yet reported [18, 27]. A nonlinear method based on neural network was proposed but an analysis of its performance was not provided [28]. Some redesigned genes were proven for high expression by experiment [19, 29–31]. However, the most important disadvantage is that almost all of these studies did not provide any method to construct the model within an actual experimental data and to evaluate the optimization methods based on statistics [16–20, 22, 27].

Machine learning approaches have been developed rapidly for recent decades. These methods could analyze and "learn" pattern from data sources and predict precisely the outcome of a new data instance. Artificial neural network (NN) and Bayesian decision are two of the most efficient and popular machine learning algorithm worldwide. NN is a strong learning technique and appropriated with both regression and classification problem. Bayesian decision is highly acclaimed due to its simplicity.

These are the reason why we propose a novel method for gene optimization base on Bayesian theory and Neural network which are the most common learning methods using probability and statistics background. We also use statistic test to evaluate and compare these methods.

Methods

Data collection

We used highly expressed genes (HEG) as the reference set for codon adaptation index (CAI) computing [32]. We also collected redesigned genes and respective translational expression levels of product (Table 1). The experimental data collection process was based on four criteria: 1) expression system should be *Escherichia coli*, 2) the experiments should express both native and optimized genes, 3) the sequences and respective quantitative productivity should be provided, and 4) expression level should be recorded or could be converted to mg/L. The data would be used to form an NN in a later step.

Codon usage bias measurements

The preference of codons is correlated with intracellular tRNA concentration in a host environment and reflects a balance necessary for expression efficiency [3, 6, 9, 15]. Translation process can be delayed when ribosomes encounter rare codons, which can cause ribosomes to detach from mRNA and abort translation [9]. Moreover, mRNA translation rate may impact the secondary structure of encoded protein in that frequently used codons tend to encode for structural elements while rare codons are associated with linkers [11, 33, 34].

CAI is one of the most popular and effective measures for quantifying codon usage bias of a gene toward a reference set of highly expressed genes [35]. Given a gene $g = \{g_1, g_2, \ldots, g_i, \ldots g_{L(g)}\}$, CAI is defined as (1)

$$CAI(g) = \left(\prod_{i=1}^{L(g)} w_{ag_i}\right)^{\frac{1}{L(g)}} \quad (1)$$

where $L(g)$ is the length of gene g counted by codon, g_i is the i^{th} codon of gene g, ac is generally a codon c coding for amino acid a. In this case, $c \equiv g_i$, w_{ac} described as (2) is the relative adaptiveness of ac, and $o_{ac}(HEG)$ is the count of ac in HEG set.

$$w_{ac} = \frac{o_{ac}(HEG)}{\max_{o_{ac}} o_{ac}(HEG)} \quad (2)$$

Relative synonymous codon usage (RSCU), which maps genes into a 59-dimensional vector space is also a common

Table 1 Collected data including redesigned genes and respective product

Host	Product	Number of genes	Reference
E. coli BL21	DNA Polymerase and scFV	62	[46]
E. coli BL21	Cystatin C	2	[12]
E. coli BL21	PEDF	2	[9]
E. coli W3110	Prochymosin	7	[11]

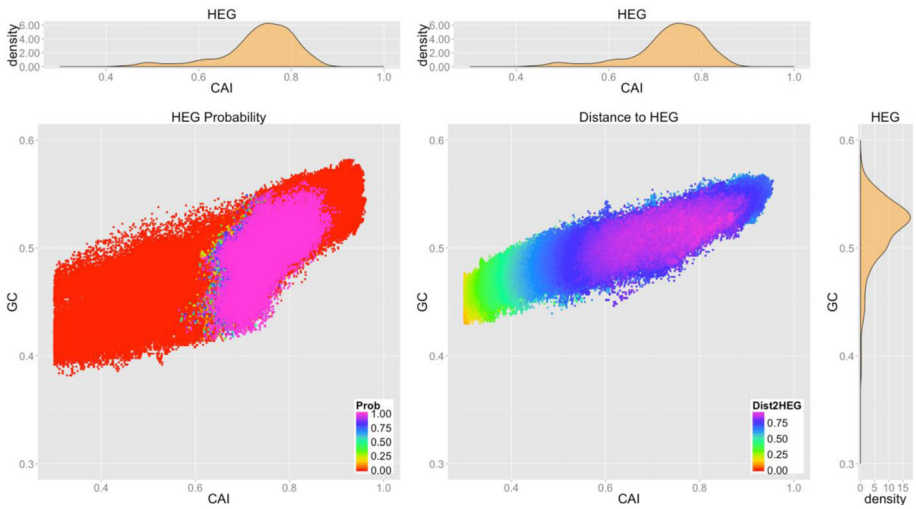

Fig. 1 Properties of HEGP and DHEG. The top plots are distribution of HEG's CAI value. The bottom right plot is distribution of HEG's GC value. The bottom center plots illustrate HEG probability and distance to HEG of randomly generated gene sequences with respect to their CAI and GC value

measure and widely used in gene clustering [36]. The RSCU is

$$r_{ac}(g) = \frac{o_{ac}(g)}{\frac{1}{k_a}\sum_{c \in C_a} o_{ac}\,(g)} \tag{3}$$

where $C_a = \{ac | ac\ is\ the\ codon\ c\ coding\ for\ amino\ acid\ a\}$, and $k_a = |C_a|$.

GC content

Some studies indicated that GC content can impact the stability of the 2nd structure of mRNA which was beneficial for translation [7, 10]. GC content is computed as (4)

$$GC(g) = \frac{o_{GC}(g)}{L(g)} \tag{4}$$

where $o_{GC}(g)$ is the count of Guanine and Cytosine in

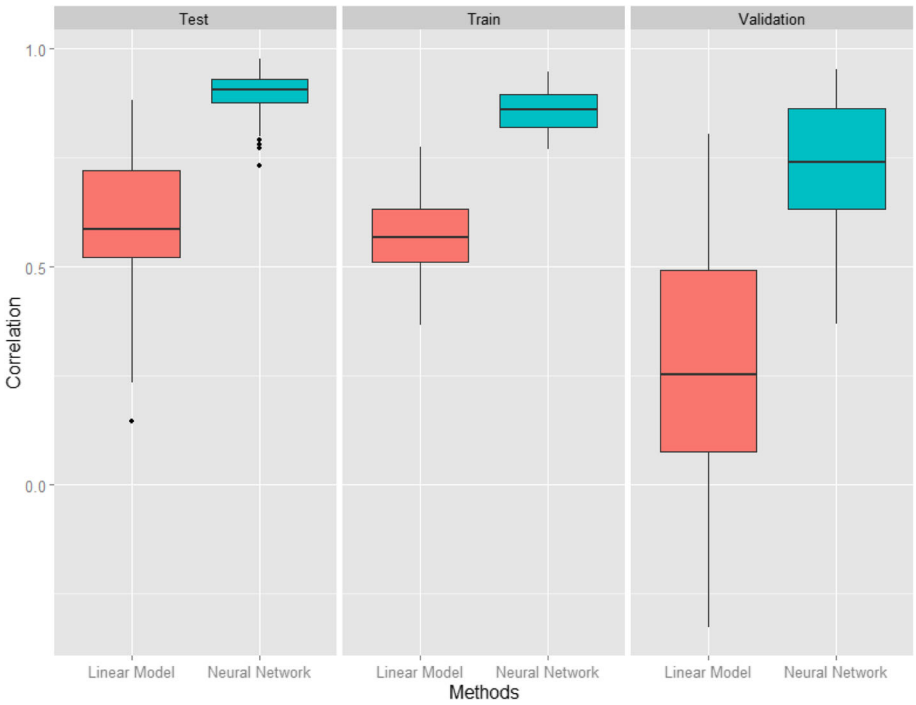

Fig. 2 Comparison between NN and linear regression

Table 2 *P*-value from Shapiro-Wilk normality test for the correlation of NN and correlation of linear regression

	Training	Validation	Testing
NN	0.03	0.00	9.10×10^{-5}
Linear regression	0.21	0.11	0.13

gene g, and $L(g)$ is the length of gene g counted by nucleotide.

Distance to HEG and HEG probability

In 2011, Menzella's research suggested that replacing all codons by the most preferable codons could lead to an inferior yield because of an imbalanced tRNA pool. Additionally, a low concentration of favorite usage codons also causes decrease in the translational level. In this case, estimating the most appropriate CAI value is an unlikely task [11, 22]. Hence, we proposed two novel features called HEG probability (HEGP) and distance to HEG (DHEG).

Given a gene g, the event that g is a member of *HEG* set or not is a random variable. The probability that g belongs to the reference set of *HEG* is shown as (5)

$$P(HEG|g) = \frac{P(g|HEG)P(HEG)}{P(g|HEG) + P\left(g|\overline{HEG}\right)} \tag{5}$$

$$P(g|HEG) = P(HEG) \times \prod_{c \in C} \frac{e^{-\frac{(r_{ac}(g)-\mu_c)^2}{2\sigma_c^2}}}{\sigma_c\sqrt{2\pi}} \tag{6}$$

$$P\left(g|\overline{HEG}\right) = P\left(\overline{HEG}\right) \times \prod_{c \in C} \frac{e^{-\frac{(r_{ac}(g)-\overline{\mu}_c)^2}{2\overline{\sigma}_c^2}}}{\overline{\sigma}_c\sqrt{2\pi}} \tag{7}$$

where c is a codon in a set of possible codon C, $\overline{HEG}$ is a non-highly expressed genes set, $\mu_c(\overline{\mu}_c)$ and $\sigma_c(\overline{\sigma}_c)$ is the mean and standard deviation of r_{ac} of all genes in *HEG* set ($\overline{HEG}$ set, respectively). In some cases that $P\left(g|\overline{HEG}\right)$ is much smaller than $P(g|HEG)$, $P(HEG|g)$ can be high although g is too different from *HEG* and $P\left(g|\overline{HEG}\right)$ is low. In order to limit this situation, we defined a new Eq. (8) limited by principle components analysis (PCA) [37, 38]

$$P_{final}(HEG|g) = \begin{cases} P(HEG|g), & p_i(g) \in [\min p_i(g_{HEG}),\ \max p_i(g_{HEG})] \\ 0, & otherwise \end{cases} \tag{8}$$

where $p_i(g)$ is the $i^{\text{th}}_{\neg}$ principle component of vector $(r_{a1}(g), \ldots, r_{a59}(g))^T$, g_{HEG} is a gene in *HEG* set, and $i = \overline{1,2}$. We only used two principle components, thus, we could observe HEGP on a 2-dimensional surface.

Resembling HEGP, DHEG was used to calculate the similarity between the candidate gene g and the reference set as (9). We normalized DHEG by scaling $D(g, HEG)$ to [0,1] such that $D(g, HEG) = 0$ if g and *HEG* are totally different, and $D(g, HEG) = 1$ if they highly resemble, see (10). In our experiment, $\min_g D(g, HEG) = 4$ and $\max_g D(g, HEG) = 17$.

$$D(g, HEG) = \frac{1}{|HEG|} \sum_{g_{HEG} \in HEG} \left\| \begin{pmatrix} r_{a1}(g) \\ \vdots \\ r_{a59}(g) \end{pmatrix} - \begin{pmatrix} r_{a1}(g_{HEG}) \\ \vdots \\ r_{a59}(g_{HEG}) \end{pmatrix} \right\|_2 \tag{9}$$

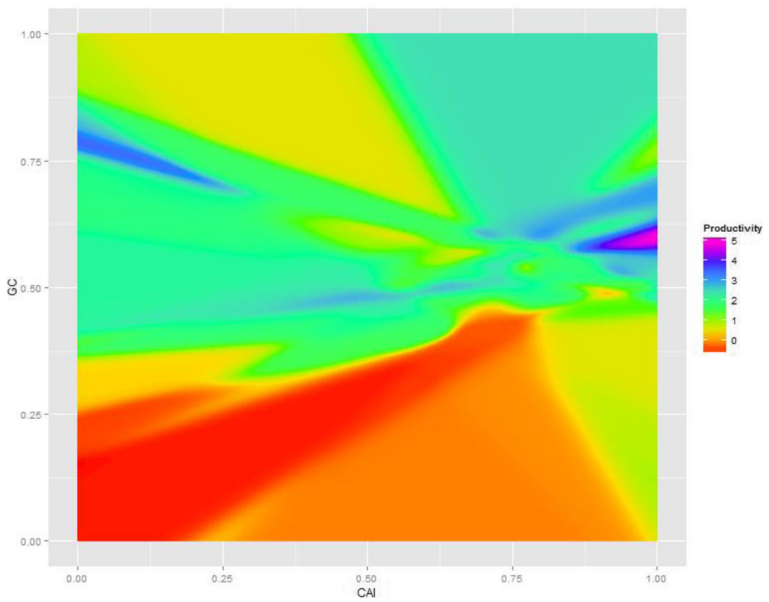

Fig. 3 Visualization for fitness function based on NN (log scale) with respect to CAI and GC value

$$D_{final}(g, HEG) = \frac{D(g, HEG) - \min_g D(g, HEG)}{\max_g D(g, HEG) - \min_g D(g, HEG)} \quad (10)$$

In order to investigate the properties of the novel features, we used a genetic algorithm to optimize genes g coding for random proteins as (11) or (12)

$$g = \arg\max_g \left[|CAI(g) - i| + |GC(g) - j| + |P_{final}(HEG|g) - k| \right] \quad (11)$$

$$g = \arg\max_g \left[|CAI(g) - i| + |GC(g) - j| + |D_{final}(g, HEG) - k| \right] \quad (12)$$

where $i, j, k \in \{0.00, 0.01, \ldots, 1.00\}$. We would like to obtain all possible value of HEGP and DHEG with respect to each pair of CAI and GC value within this process and analyze the association between CAI and GC and HEGP (or DHEG). The results are shown in Results and Discussion and Fig. 1.

Neural network

We proposed a novel method to construct fitness function for genetic algorithm (in next step) based on neural network (NN), CAI and GC content. A 2-hidden layer network is computed as (17), such that $\Sigma_g |o(g) - y_g|^2$ was minimized, where y_g is the yield of gene g collected from experimental data (in Data collection), m is the number of nodes at the first hidden layer, and n is the number of nodes at the second hidden layer [39]. We estimated $m = \sqrt{3N} + 2\sqrt{\frac{N}{3}}$ and $n = 2\sqrt{\frac{N}{3}}$ as Huang suggested in 2003, where N is number of samples in data set [40].

$$o_1(g) = \begin{pmatrix} w^1_{1,1} & \cdots & w^1_{1,3} \\ \vdots & \ddots & \vdots \\ w^1_{m,1} & \cdots & w^1_{m,3} \end{pmatrix} \times \begin{pmatrix} CAI(g) \\ GC(g) \\ 1 \end{pmatrix} \quad (13)$$

$$h_1(g) = \frac{1}{1 + e^{-o_1}} \quad (14)$$

$$o_2(g) = \begin{pmatrix} w^2_{1,1} & \cdots & w^2_{1,m+1} \\ \vdots & \ddots & \vdots \\ w^2_{n,1} & \cdots & w^2_{n,m+1} \end{pmatrix} \times \begin{pmatrix} h_1 \\ 1 \end{pmatrix} \quad (15)$$

$$h_2(g) = \frac{1}{1 + e^{-o_2}} \quad (16)$$

$$o(g) = \begin{pmatrix} w_1 & \cdots & w_{n+1} \end{pmatrix} \times \begin{pmatrix} h_2 \\ 1 \end{pmatrix} \quad (17)$$

For the purpose of testing performance of this method, we randomly separated data into 3 parts which were 30% of data for testing, 70% × 30% = 21% of data for validation, and 70% × 70% = 49% of data for training. Training, validation, and testing processed were repeated 100 times to reduce impact of over fitting, and the final model was an arithmetic mean of these 100 NNs, (18).

$$o_{final} = \frac{1}{100} \sum_{i=1}^{100} o^{(i)}(g) \quad (18)$$

We also restricted NN by HEGP (or DHEG) such that $o_{final} = P_{final}(HEG|g)$ (or $o_{final} = D_{final}(g, HEG)$) if c (or $D_{final}(g, HEG) < 0.75$), otherwise o_{final} is considered as (18). These were called NN restricted by HEGP (NNP) and NN restricted by DHEG (NND).

Multivariable linear regression

Linear functions were commonly used in gene optimization [18, 27], as (19). In this study, we proposed estimating

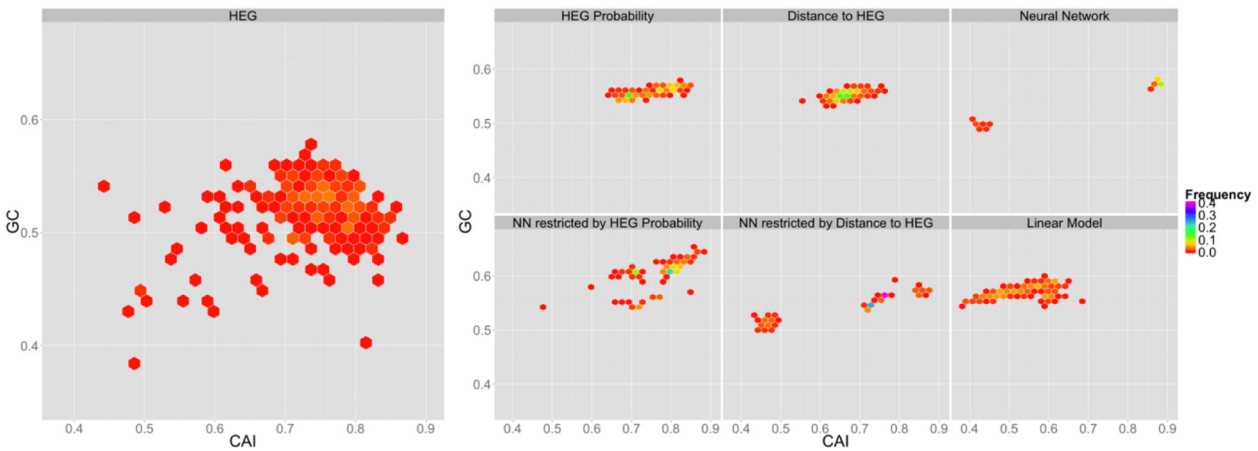

Fig. 4 Comparison between gene optimization methods. The plots are 2-dimensional distribution of redesigned genes. X and Y coordinate are CAI and GC value, respectively

Table 3 Descriptive statistics for optimized genes

	Statistics	HEG	HEGP	DHEG	NN	NNP	NND	Linear regression
CAI	Min	0.44	0.64	0.64	0.38	0.47	0.42	0.36
	1st Quantile	0.70	0.70	0.70	0.89	0.72	0.72	0.47
	Median	0.75	0.75	0.75	0.90	0.80	0.76	0.54
	Mean	0.73	0.75	0.75	0.88	0.78	0.71	0.53
	3rd Quantile	0.79	0.80	0.80	0.90	0.81	0.76	0.59
	Max	0.86	0.85	0.85	0.92	0.88	0.86	0.68
GC	Min	0.39	0.54	0.54	0.48	0.54	0.50	0.54
	1st Quantile	0.50	0.55	0.55	0.58	0.61	0.54	0.56
	Median	0.52	0.56	0.56	0.58	0.61	0.56	0.57
	Mean	0.52	0.56	0.56	0.58	0.60	0.54	0.57
	3rd Quantile	0.54	0.56	0.56	0.58	0.62	0.56	0.58
	Max	0.58	0.58	0.58	0.58	0.65	0.59	0.60

The orange cells represent for values, which are different more than 5% from values of HEG, and vice versa for green cells

parameters such that $\sum_g \left(y_g - \widehat{y_g}\right)^2$ was minimized [41]. We also separated data as Neural network for comparison purposes and the final model was constructed by using whole data set.

$$\widehat{y_g} = \left(\widehat{w_1} \quad \widehat{w_2}\right) \times \begin{pmatrix} CAI(g) \\ GC(g) \end{pmatrix} + \hat{\epsilon} \quad (19)$$

Genetic algorithm

The genetic algorithm which was inspired by natural selection and evolution processes is naturally appropriate to the gene optimization task in that each gene was assigned as a chromosome or an individual [6, 42, 43]. These genes could be evaluated by a fitness function which are $P_{final}(\text{-}HEG|g)$, $D_{final}(g, HEG)$, $o_{final}(g)$ or $\widehat{y_g}$. Generation to generation, the algorithm would converge and reach the maximum value of fitness function. Finally, we found the best gene with respect to the fitness function.

Results and Discussion

Properties of HEGP and DHEG

Figure 1 illustrates the distributions of HEGP and DHEG in the 2-dimensional vector space constructed by CAI and GC content value. As Fig. 1 described, HEGP of genes varies from 0.00 to 1.00. However, because of the PCA technique, genes having high HEGP tend to cluster together and separate completely from the other genes having minimum HEGP by a discriminant boundary.

DHEG also varies from 0.00 to 0.70, but there is no boundary separating regions of high and low DHEG. Genes having high HEGP or DHEG seem to distribute in the region of high CAI and GC content density. This result suggests that both HEGP and DHEG associate with HEG set in CAI and GC content aspects.

Properties of NN and comparison between NN and linear regression

In comparison with linear regression method, correlation of NN is 1.50, 2.69, and 1.50 times higher than that of linear regression within training, validation, and testing processes, respectively, Fig. 2. A Shapiro-Wilk test shows that almost all data do not fit a normal distribution, so we used a non-parametric Wilcoxon signed-rank test to investigate whether there is any significant difference between the correlation given by NN and linear model, Table 2 [44]. The test indicates that correlation coefficients from NN are significantly higher than that given by linear regression (*P-value* $< 2.2 \times 10^{-6}$ for both three processes) [45]. This result suggests that NN is much more accurate than linear regression. In fact, most of phenomena and processes in nature, especially in life science, associate with non-linear models. For instance, both population growth, gene expression, epidemic spread, etc. models are fitted well with non-linear models. This is a reasonable explanation for the high performance of NN, a non-linear model with sigmoid function.

However, NN usually faces an over fitting problem, which causes inaccuracy in practice. As shown in Fig. 3,

Table 4 *P*-value from Shapiro-Wilk normality test for optimized genes

	HEG	HEGP	DHEG	NN	NNP	NND	Linear regression
CAI	1.26×10^{-10}	1.06×10^{-7}	0.00	$< 2.2 \times 10^{-16}$	2.06×10^{-12}	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$
GC	2.66×10^{-9}	0.01	0.19	$< 2.2 \times 10^{-16}$	2.73×10^{-16}	4.20×10^{-12}	4.20×10^{-12}

Table 5 *P*-values from Wilcoxon signed-rank test for difference between HEG and optimized genes

	HEGP	DHEG	NN	NNP	NND	Linear regression
CAI	0.29	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	2.37×10^{-12}	0.26	$< 2.2 \times 10^{-16}$
GC	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$

there are some unreasonable local maximum regions such as the blue region on the top-left corner, and the purple region on the middle-right of the figure. Although genes in these regions have been reported to have low productivity, they still are predicted to have a high translational level. This is the result of a small data set and the complexity of the NN. To overcome this situation, we modified HEGP as in Distance to HEG and HEG probability and the result is shown in Comparison between proposed methods and Application for Escherichia coli to compare between optimization methods.

Comparison between proposed methods

We also optimized genes in HEG to compare gene optimization methods and the results are visualized in Fig. 4. While HEGP and DHEG highly appropriate with HEG (differences in descriptive statistics values do not exceed 5% as represented by green cells in Table 3), genes redesigned by linear regression method locate in the region of low CAI (from 0.36 and 0.68) and are quite different from HEG (orange cells in Table 3). NNP is also potential for gene optimization, but NN and NND seem to be unstable and a part of genes optimized by these two methods locate in low CAI region because of the over fitting problem. Both NNP and distances to HEG are the same with HEG, but NN are more than 5% different from HEG. All data in this experiment are not under a normal distribution (Table 4) and the Wilcoxon signed-rank test shows that genes redesigned by HEGP and NND resemble HEG ($P\text{-value} > 0.05$), whereas genes optimized by DHEG, NN, NNP and linear model are significantly different from HEG, regarding CAI (Table 5). The distribution of genes designed by NN and linear model are different form HEG so it is reasonable that $P\text{-value} < 0.05$ in these cases. Although NNP and DHEG seem to be associated with HEG, the test shows that genes designed by these methods are different from HEG because these methods only focus on high density region of HEG.

Application for *Escherichia coli* to compare between optimization methods

We also redesigned gene coding for prochymosin, which was well optimized by Menzella to introduce to *Escherichia coli* in 2011, in order to compare with JCat and EuGene programs [11, 18, 19]. Menzella's study suggested that CAI of HEG coding for prochymosin are from 0.70 to 0.74 and CAI of the gene giving highest yield is 0.72. Genes having CAI that is out of that range were reported as to be low expressed. In this study, we used the best gene of Menzella's study as the criteria to evaluate and compare gene optimization method. As Table 6 described, all redesigned genes give the same GC content as the one of Menzella. Only DHEG gives CAI in highly expressed range (0.73) and the CAI value is just lower than CAI of the standard genes by 1.39%, whereas the ones from NN are 34.72% lower than the criteria of CAI. CAI from JCat, EuGene, and linear model are considerably different from the standard by 33.33, 30.56, and 27.78%, respectivly. There are just slightly differences, which are 12.50, 6.94, and 11.11% between the best gene from Menzella's study and genes optimized by HEGP, NNP, and NND. Gene

Table 6 Result of optization for gene coding for prochymosin and comparison with experimental result from Menzella's study

Method	CAI	GC	Patterns matching				
Menzella	0.72	0.49	6 nucleotides	7 nucleotides	8 nucleotides	9 nucleotides	Total
Jcat	0.96	0.50	177	69	22	12	280
Eugene	0.94	0.50	48	20	6	2	76
HEGP	0.81	0.51	173	55	15	6	249
DHEG	0.73	0.50	216	61	13	1	291
NN	0.47	0.49	185	57	20	7	269
NNP	0.67	0.50	204	68	26	13	311
NND	0.64	0.50	198	63	17	4	282
Linear	0.52	0.52	185	64	13	2	264

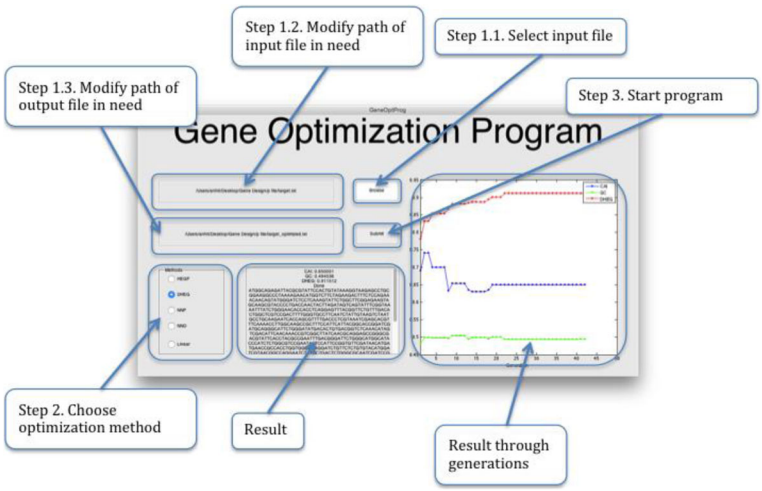

Fig. 5 Demonstration program and user's guide

redesigned by NND is most matched with Menzella's gene (311), while EuGene gives the worst (67). Matching results of other methods are roughly the same, from 249 to 291. According to these results, we can indicate that EuGene and NN seem not to appropriate to redesign prochymochin. Although matching result from JCat is the fourth highest (280), JCat is also inappropriate because of the high CAI value (0.96). DHEG is likely the most appropriated method with reasonable CAI (0.73) and high matching result (291). HEGP, NNP, and NND give CAI values, which are slightly different from Menzella's result, but these are also potential methods because genes optimized by these methods highly match with the best gene of Menzella.

Finally, we built a demonstrative program using Matlab R2014a under Mac OS X. The program was published in both standalone executable program and Matlab function files. As in Fig. 5, gene optimization includes 3 steps:

Step 1. Select target protein sequences in FASTA format
Step 2. Choose optimization method
Step 3. Start program.

While the program run, the text box on the bottom and the chart on the right will illustrate the progress. The result will be presented in the text box and also stored as FASTA format. The developed program can be downloaded from http://www.math.hcmus.edu.vn/~ptbao/paper_soft/GeneOptProg/.

There are limitations of our study. Firstly, HEG reference set for CAI computing is obtained from predictive method with no laboratory evidence showing that the set is actual HEG dataset. Other studies share the same problem, but the redesigned gene based on CAI computation with predicted HEG are highly expressed [19, 29–31]. Secondly, NN is sensitive in that a small change of input can lead to a significant change of output and it also tends to over fit the training data. Data is collected from different sources with a limited number of samples under variety of experimental environments. These contributes to over-fitting of the NN method. Lastly, although the optimized genes closely resemble highly expressed redesigned genes in related studies, results of proposed methods are not verified by wet lab experiments.

Conclusions

In this study, we proposed the uses of HEGP, DHEG, and NN to optimize genes and also indicated an approach to estimate parameters for linear function in gene optimization. The correlations of our proposed NN method are from 1.5 to 2.69 times greater than these of linear regression method. Additionally, genes redesigned by the proposed methods associate with HEG whereas genes optimized by popular linear function give low CAI. Therefore, it is concluded that our proposed methods can be potential for gene optimization and further research in gene expression.

In the future, more redesigned genes will be collected to enrich our database to improve the performance of NN. In addition, a mathematical model based on differential equation will be developed to investigate how codon usage bias and tRNA concentration influence translation expression level. The developed model can then be applied in gene optimization. Finally, experiment will be carried out to test the proposed methods and hypothesis.

Abbreviations
C_a: All different types of codon encoding for a specific amino acid a; $D_{final}(g, HEG)$: Normalized distance between gene g and HEG; $P_{final}(HEG|g)$: Probability that g belongs to the reference set of HEG, limited by PCA; $k_a = |C_a|$: Number of different types of codon encoding for a amino acid a; $o_{GC}(g)$: Count of Guanine and Cytosine in gene g; $o_{ac}(HEG)$: Count of codon c coding for amino acid a in HEG set; o_{final}: Average output of 100 NNs; $r_{ac}(g)$: RSCU for codon c coding for amino acid a; w_{ac}: Relative adaptiveness of codon c coding for amino acid a; CAI: Codon adaptation index; DHEG: Distance to HEG; HEG: Highly expressed genes/gene; HEGP: HEG probability; NN: Neural network; NND: NN restricted by DHEG; NNP: NN restricted by HEGP; PCA: Principle components analysis; RSCU: Relative synonymous codon usage; $D(g, HEG)$: Distance between gene g and HEG; $GC(g)$: GC content of gene g; $L(g)$: Length of gene g counted by codon/nucleotide; $P(HEG|g)$: Probability that g belongs to the reference set of HEG; g: A gene; $g(i)$: The i^{th} codon of gene g; $o(g)$: Output of NN

Acknowledgements
We would like to thank Dr. Nguyen Duc Hoang, Msc. Vo Tri Nam, Mr. Truong Vo Huu Thien, Mr. Nguyen Duy Tung, and Ms. Dang Thi Hang for their support and advice. We would like to thank Welch et. al, Wang et. al, Gvritishvili et. al, and Menzella et. al for providing data.

Funding
Not applicable.

Availability of data and materials
The program supporting the conclusion of this article is available in http://www.math.hcmus.edu.vn/~ptbao/paper_soft/GeneOptProg/.

Authors' contributions
Collected data, designed methods, conducted experiments, and wrote paper: TT-A. Involved in discussions, analysis plans for the manual script from its inception, including the idea of the data analysis, drafting it, and revising it critically: LTL, NQV, and PTB. All authors read and approved the final manual script.

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Not applicable.

Ethics approval and consent to participate
Not applicable.

Author details
[1]Faculty of Mathematics and Computer Science, VNUHCM-University of Science, 227 Nguyen Van Cu Street, District 5, Ho Chi Minh City, Vietnam. [2]School of Biotechnology, VNUHCM-International University, Quarter 6, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam. [3]Faculty of Information Technology, Ho Chi Minh City University of Pedagogy, 280 An Duong Vuong Street, Ward 4, District 5, Ho Chi Minh City, Vietnam.

Received: 23 April 2016 Accepted: 1 February 2017
Published online: 10 February 2017

References

1. Balbás P, Lorence A. Recombinant gene expression: reviews and protocols. Totowa: Humana Press; 2004.
2. Elena C, Ravasi P, Castelli ME, Peirú S, and Menzella HG. "Expression of codon optimized genes in microbial systems: current industrial applications and perspectives." Front Microbiol 2014;vol. 5.
3. Gouy M, Gautier C. Codon usage in bacteria: correlation with gene expressivity. Nucleic Acids Res. 1982;10(22):7055–74.
4. Gustafsson C, Govindarajan S, Minshull J. Codon bias and heterologous protein expression. Trends Biotechnol. 2004;22(7):346–53.
5. Henry I, Sharp PM. Predicting gene expression level from codon usage bias. Mol Biol Evol. 2006;24(1):10–2.
6. Sandhu KS, Pandey S, Maiti S, Pillai B. GASCO: Genetic Algorithm Simulation for Codon Optimization. In Silico Biol. 2008;8(2):187–92.
7. Wu X, Jörnvall H, Berndt KD, Oppermann U. Codon optimization reveals critical factors for high level expression of two rare codon genes in Escherichia coli: RNA stability and secondary structure but not tRNA abundance. Biochem Biophys Res Commun. 2004;313(1):89–96.
8. Bai J, Swartz DJ, Protasevich II, Brouillette CG, Harrell PM, Hildebrandt E, Gasser B, Mattanovich D, Ward A, Chang G, Urbatsch IL. A gene optimization strategy that enhances production of fully functional P-glycoprotein in Pichia pastoris. PLoS ONE. 2011;6(8):e22577.
9. Gvritishvili AG, Leung KW, Tombran-Tink J. Codon preference optimization increases heterologous PEDF expression. PLoS ONE. 2010;5(11):e15056.
10. Li W, Ng I-S, Fang B, Yu J, Zhang G. Codon optimization of 1,3-propanediol oxidoreductase expression in Escherichia coli and enzymatic properties. Electron J Biotechnol. 2011;14(4).
11. Menzella HG. Comparison of two codon optimization strategies to enhance recombinant protein production in Escherichia coli. Microb Cell Factories. 2011;10(1):15.
12. Wang Q, Mei C, Zhen H, Zhu J. Codon preference optimization increases prokaryotic Cystatin C expression. J Biomed Biotechnol. 2012;2012:1–7.
13. Grantham R, Gautier C, Gouy M, Mercier R, Pavé A. Codon catalog usage and the genome hypothesis. Nucleic Acids Res. 1980;8(1):r49–62.
14. Ikemura T. Codon usage and tRNA content in unicellular and multicellular organisms. Mol Biol Evol. 1985;2(1):13–34.
15. Ikemura T. Correlation between the abundance of Escherichia coli transfer RNAs and the occurrence of the respective codons in its protein genes. J Mol Biol. 1981;146(1):1–21.
16. Fuglsang A. Codon optimizer: a freeware tool for codon optimization. Protein Expr Purif. 2003;31(2):247–9. Tháng Mư i.
17. Gao W, Rzewski A, Sun H, Robbins PD, Gambotto A. UpGene: application of a Web-based DNA codon optimization algorithm. Biotechnol Prog. 2004;20(2):443–8. Tháng M t.
18. Gaspar P, Oliveira JL, Frommlet J, Santos MAS, Moura G. EuGene: maximizing synthetic gene design for heterologous expression. Bioinformatics. 2012;28(20):2683–4.
19. Grote A, Hiller K, Scheer M, Münch R, Nörtemann B, Hempel DC, Jahn D. JCat: a novel tool to adapt codon usage of a target gene to its potential expression host. Nucleic Acids Res. 2005;33 suppl 2:W526–31.
20. Hoover DM. DNAWorks: an automated method for designing oligonucleotides for PCR-based gene synthesis. Nucleic Acids Res. 2002;30(10):43e–43.
21. Supek F, Vlahoviček K. INCA: synonymous codon usage analysis and clustering by means of self-organizing map. Bioinformatics. 2004;20(14):2329–30.
22. Villalobos A, Ness JE, Gustafsson C, Minshull J, Govindarajan S. Gene Designer: a synthetic biology tool for constructing artificial DNA segments. BMC Bioinformatics. 2006;7(1):285.
23. Guo Y, Wallace SS, Bandaru V. A novel bicistronic vector for overexpressing Mycobacterium tuberculosis proteins in Escherichia coli. Protein Expr Purif. 2009;65(2):230–7.
24. Rosano GL, Ceccarelli EA. Rare codon content affects the solubility of recombinant proteins in a codon bias-adjusted Escherichia coli strain. Microb Cell Factories. 2009;8(1):41.
25. Widmann M, Clairo M, Dippon J, Pleiss J. Analysis of the distribution of functionally relevant rare codons. BMC Genomics. 2008;9(1):207.
26. Zhou Z, Schnake P, Xiao L, Lal AA. Enhanced expression of a recombinant malaria candidate vaccine in Escherichia coli by codon optimization. Protein Expr Purif. 2004;34(1):87–94.
27. Raab D, Graf M, Notka F, Schödl T, Wagner R. The GeneOptimizer Algorithm: using a sliding window approach to cope with the vast sequence space in multiparameter DNA sequence optimization. Syst Synth Biol. 2010;4(3):215–25.
28. Jung S-K, McDonald K. Visual gene developer: a fully programmable bioinformatics software for synthetic gene optimization. BMC Bioinformatics. 2011;12(1):340.
29. Solis-Escalante D, Kuijpers NGA, Bongaerts N, Bolat I, Bosman L, Pronk JT, Daran J-M, Daran-Lapujade P. *amdSYM*, a new dominant recyclable marker cassette for *Saccharomyces cerevisiae*. FEMS Yeast Res. 2013;13(1):126–39.
30. Blackburn MC, Petrova E, Correia BE, Maerkl SJ. Integrating gene synthesis and microfluidic protein analysis for rapid protein engineering. Nucleic Acids Res. 2016;44(7):e68.
31. Stammen S, Muller BK, Korneli C, Biedendieck R, Gamer M, Franco-Lara E, Jahn D. High-yield intra- and extracellular protein production using bacillus megaterium. Appl Environ Microbiol. 2010;76(12):4037–46.
32. Tuan-Anh T, Nam V, Hoang N, Bao P. "A Novel Method to Highly Expressed Genes Prediction Using Radius Clustering and Relative Synonymous Codon Usage." J Comput Biol. 2015;1086–96.

33. Marin M. Folding at the rhythm of the rare codon beat. Biotechnol J. 2008;3(8):1047–57.
34. Purvis IJ, Bettany AJE, Santiago TC, Coggins JR, Duncan K, Eason R, Brown AJP. The efficiency of folding of some proteins is increased by controlled rates of translation in vivo. J Mol Biol. 1987;193(2):413–7.
35. Sharp PM, Li W-H. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Res. 1987;15(3):1281–95.
36. Sharp PM, Tuohy TMF, Mosurski KR. Codon usage in yeast: cluster analysis clearly differentiates highly and lowly expressed genes. Nucleic Acids Res. 1986;14(13):5125–43.
37. Duda RO, Hart PE, Stork DG. Pattern classification. 2nd ed. New York: Wiley; 2001.
38. Jolliffe IT. Principal Component Analysis. New York: Springer New York; 1986.
39. Mitchell TM. Machine Learning. 1st ed. New York: McGraw-Hill, Inc.; 1997.
40. Huang G-B. Learning capability and storage capacity of two-hidden-layer feedforward networks. IEEE Trans Neural Netw. 2003;14(2):274–81. Tháng Ba.
41. Yan X. Linear Regression Analysis: Theory and Computing, World Scientific. 2009.
42. Michalewicz Z. Genetic algorithms + data structures = evolution programs, 3rd rev. and extended ed. Berlin. New York: Springer; 1996.
43. Mitchel M. An introduction to genetic algorithms. Cambridge: MIT Press; 1996.
44. Royston JP. An extension of Shapiro and Wilk's W test for normality to large samples. Appl Stat. 1982;31(2):115.
45. Hollander M, Wolfe DA. Nonparametric statistical methods. 2nd ed. New York: Wiley; 1999.
46. Welch M, Govindarajan S, Ness JE, Villalobos A, Gurney A, Minshull J, Gustafsson C. Design parameters to control synthetic gene expression in Escherichia coli. PLoS ONE. 2009;4(9):e7002.